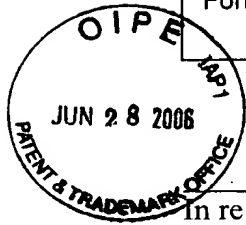


HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

Docket No.: 10013499-1
(PATENT)



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Joubert Berger et al.

Application No.: 09/896,385

Confirmation No.: 9535

Filed: June 29, 2001

Art Unit: 2195

For: SYSTEM AND METHOD FOR
MANAGEMENT OF COMPARTMENTS IN A
TRUSTED OPERATING SYSTEM

Examiner: K. Tang

APPEAL BRIEF

MS Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

As required under § 41.37(a), this brief is filed within two months of the Notice of Appeal filed in this case on April 28, 2006, and is in furtherance of said Notice of Appeal.

The fees required under § 41.20(b)(2) are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

This brief contains items under the following headings as required by 37 C.F.R. § 41.37 and M.P.E.P. § 1206:

- | | |
|-------|---|
| I. | Real Party In Interest |
| II | Related Appeals and Interferences |
| III. | Status of Claims |
| IV. | Status of Amendments |
| V. | Summary of Claimed Subject Matter |
| VI. | Grounds of Rejection to be Reviewed on Appeal |
| VII. | Argument |
| VIII. | Claims Appendix |

- IX. Evidence Appendix
- X. Related Proceedings Appendix

I. REAL PARTY IN INTEREST

The real party in interest for this appeal is:

Hewlett-Packard Development Company, L.P., a Limited Partnership established under the laws of the State of Texas and having a principal place of business at 20555 S.H. 249, Houston, TX 77070, U.S.A. (hereinafter "HPDC"). HPDC is a Texas limited partnership and is a wholly-owned affiliate of Hewlett-Packard Company, a Delaware Corporation, headquartered in Palo Alto, CA. The general or managing partner of HPDC is HPQ Holdings, LLC.

II. RELATED APPEALS, INTERFERENCES, AND JUDICIAL PROCEEDINGS

There are no other appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

III. STATUS OF CLAIMS

A. Total Number of Claims in Application

There are 28 claims pending in application.

B. Current Status of Claims

- 1. Claims canceled: claim number 2
- 2. Claims withdrawn from consideration but not canceled: None
- 3. Claims pending: 1 and 3-29
- 4. Claims allowed: None
- 5. Claims rejected: 1 and 3-29

C. Claims On Appeal

The claims on appeal are claims 1 and 3-29.

IV. STATUS OF AMENDMENTS

A Final Office Action rejecting the claims of the present application was mailed March 13, 2006. In response, Applicant did not file an Amendment After Final Rejection, but instead filed a Notice of Appeal, which this brief supports. Accordingly, the claims on appeal are those as rejected in the Final Office Action of March 13, 2006. A complete listing of the claims is provided in the Claims Appendix hereto.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The following provides a concise explanation of the subject matter defined in each of the claims involved in the appeal, referring to the specification by page and line number and to the drawings by reference characters, as required by 37 C.F.R. § 41.37(c)(1)(v). Each element of the claims is identified by a corresponding reference to the specification and drawings where applicable. It should be noted that the citation to passages in the specification and drawings for each claim element does not imply that the limitations from the specification and drawings should be read into the corresponding claim element.

According to one claimed embodiment, such as that of independent claim 1, a method of administering a processor-based system comprises implementing, by an operating system (e.g., operating system 101 of FIGURE 1), at least one compartment (e.g., compartment A and/or compartment B of FIGURE 3) for containment of at least one process (e.g., processes X, Y, and/or Z of FIGURE 3) executable on the processor-based system (*see* page 8, lines 1-6, page 11, lines 1-17, and page 12, line 16 – page 15, line 16 of the specification). The at least one compartment defines whether the at least one process contained therein is allowed access to particular system resources (e.g., resources A, B, and/or C of FIGURE 3, and *see* page 12, line 16 – page 15, line 16 of the specification). The method further comprises providing, by the processor-based system, at least one operating system command-line utility (e.g., command-line utilities 404 of FIGURE 4) executable to manipulate the at least one compartment (*see* page 8, lines 1-4, page 11, lines 18-24, and page 15, line 17 – page 17, line 13 of the specification).

In certain embodiments, such as that of claim 5, the implementing step comprises defining the at least one compartment in at least one configuration file (*see* page 8, lines 1-9, page 29, lines 2-9, and page 30, lines 4-13 of the specification). Further, in certain

embodiments, such as that of claim 6, the at least one command-line utility is executable to manipulate the at least one compartment without requiring a user to edit the at least one configuration file (*see* page 31, lines 16-24 of the specification).

In certain embodiments, such as that of claim 7, the implementing step comprises providing at least one rule that defines containment of the at least one compartment in at least one configuration file (*see* page 8, lines 1-9, page 29, lines 2-9, and page 30, lines 4-13 of the specification). Further, in certain embodiments, such as that of claim 8, the method further comprises providing at least one command-line utility executable to manipulate the at least one rule (*see* page 31, lines 16-24 of the specification).

In certain embodiments, such as that of claim 9, the at least one command-line utility executable to manipulate the at least one rule comprises at least one command-line utility executable to perform at least one type of rule manipulation selected from the group consisting of: adding a new rule for a particular compartment, removing an existing rule for a particular compartment, and listing all rules for a particular compartment (*see* page 34, line 1 – page 35, line 14 of the specification).

In certain embodiments, such as that of claim 25, the implementing at least one compartment comprises: utilizing a kernel (e.g., the kernel of FIGURES 4 and 5) for enforcing the at least one compartment (*see* page 12, lines 16-22 of the specification).

According to another claimed embodiment, such as that of independent claim 10, a system comprises an operating system (e.g., operating system 101 of FIGURE 1) stored to computer-readable medium. The operating system implements at least one compartment (e.g., compartment A and/or compartment B of FIGURE 3) to which at least one process (e.g., processes X, Y, and/or Z of FIGURE 3) executable on the system can be associated (*see* page 8, lines 1-6, page 11, lines 1-17, and page 12, line 16 – page 15, line 16 of the specification). The system further comprises at least one configuration file stored to computer-readable medium, the at least one configuration file defining the at least one compartment (*see* page 8, lines 1-9, page 29, lines 2-9, and page 30, lines 4-13 of the specification). The system further comprises means (e.g., command-line utilities 404 of FIGURE 4) for performing management of the at least one compartment without requiring that a user edit the at least one configuration file in which the at least one compartment is

defined (*see* page 8, lines 1-4, page 11, lines 18-24, and page 15, line 17 – page 17, line 13 of the specification).

In certain embodiments, such as that of claim 19, the means for performing management comprises at least one operating system command-line utility executable to manage the at least one compartment (*see* page 8, lines 1-4, page 11, lines 18-24, and page 15, line 17 – page 17, line 13 of the specification).

According to another claimed embodiment, such as that of independent claim 20, a computer-readable medium including instructions executable by a processor is provided. The computer-readable medium comprises a library of software functions (*see* page 8, lines 9-13 of the specification) for managing at least one compartment (e.g., compartment A and/or compartment B of FIGURE 3) implemented by an operating system (e.g., operating system 101 of FIGURE 1), wherein at least one process (e.g., processes X, Y, and/or Z of FIGURE 3) is associated with the at least one compartment and the at least one compartment defines accessibility of resources (e.g., resources A, B, and/or C of FIGURE 3) for the at least one process associated therewith (*see* page 8, lines 1-6, page 11, lines 1-17, and page 12, line 16 – page 15, line 16 of the specification). The library of software functions includes at least one command-line utility (e.g., command-line utilities 404 of FIGURE 4) executable to manipulate the at least one compartment (*see* page 8, lines 1-4, page 11, lines 18-24, and page 15, line 17 – page 17, line 13 of the specification).

According to another claimed embodiment, such as that of independent claim 26, a system comprises an operating system (e.g., operating system 101 of FIGURE 1) stored to computer-readable medium. The operating system implements at least one compartment (e.g., compartment A and/or compartment B of FIGURE 3) to which at least one process (e.g., processes X, Y, and/or Z of FIGURE 3) executable on the system can be associated. The system further comprises at least one configuration file stored to computer-readable medium, the at least one configuration file defining said at least one compartment (*see* page 8, lines 1-9, page 29, lines 2-9, and page 30, lines 4-13 of the specification). The system further comprises a command-line utility (e.g., command-line utilities 404 of FIGURE 4) executable for performing management of the at least one compartment without requiring that a user edit the at least one configuration file in which the at least one compartment is defined

(see page 8, lines 1-4, page 11, lines 18-24, and page 15, line 17 – page 17, line 13 of the specification).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1, 3, 5, and 7-9 are rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,850,511 issued to Stoecker et al. (hereinafter “*Stoecker*”) in view of U.S. Patent No. 6,449,643 issued to Hyndman et al. (hereinafter “*Hyndman*”).

Claim 4 is rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Hyndman* and further in view of U.S. Patent No. 5,930,154 issued to Thalhammer-Reyero (hereinafter “*Thalhammer-Reyero*”).

Claim 6 is rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Hyndman* and further in view of U.S. Patent No. 6,493,751 issued to Tate et al. (hereinafter “*Tate*”).

Claims 10, 12, 19, and 26-27 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Tate*.

Claims 11, 14, and 29 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Tate* and further in view of U.S. Patent No. 6,009,274 issued to Fletcher et al. (hereinafter “*Fletcher*”).

Claims 13, 15, and 28 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Tate* and further in view of *Thalhammer-Reyero*.

Claims 16-18 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Tate* and further in view of *Hyndman*.

Claims 20, 22, and 24 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Hyndman* in view of *Stoecker*.

Claim 21 is rejected under 35 U.S.C. § 103(a) as being unpatentable over *Hyndman* in view of *Stoecker* and further in view of *Thalhammer-Reyero*.

Claim 23 is rejected under 35 U.S.C. § 103(a) as being unpatentable over *Hyndman* in view of *Stoecker* and further in view of *Tate*.

Claim 25 is rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Hyndman* and further in view of U.S. Patent No. 6,023,765 issued to Kuhn (hereinafter “*Kuhn*”).

VII. ARGUMENT

Appellant respectfully traverses the outstanding rejections of the pending claims, and requests that the Board reverse the outstanding rejections in light of the remarks contained herein. The claims do not stand or fall together. Instead, Appellant presents separate arguments for various independent and dependent claims. Each of these arguments is separately argued below and presented with separate headings and sub-heading as required by 37 C.F.R. § 41.37(c)(1)(vii).

A. Rejections under 35 U.S.C. §103(a) over *Stoecker* in view of *Hyndman*

Claims 1, 3, 5, and 7-9 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Hyndman*. Appellant respectfully traverses these rejections below.

To establish a prima facie case of obviousness, three basic criteria must be met. *See* M.P.E.P. § 2143. First, there must be some suggestion or motivation, either in the applied references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the applied references must teach or suggest all the claim limitations. Without conceding the first or second criteria, Appellant respectfully asserts that the applied combination of *Stoecker* in view of *Hyndman* fails to teach or suggest all of the claim limitations, as discussed further below.

Independent Claim 1 and Dependent Claims 3, 5, and 7

Independent claim 1 recites:

A method of administering a processor-based system, said method comprising:
implementing, by an operating system, at least one compartment for

containment of at least one process executable on said processor-based system, wherein said at least one compartment defines whether said at least one process contained therein is allowed access to particular system resources; and

providing, by said processor-based system, at least one operating system command-line utility executable to manipulate said at least one compartment.

The Final Office Action asserts that *Stoecker* teaches the above elements except for the “at least one compartment defines whether said at least one process contained therein is allowed access to particular system resources”. However, the Office Action asserts that *Hyndman* teaches this element of claim 1, citing to the abstract of *Hyndman*, see page 3 of the Final Office Action. Appellant disagrees, as discussed below.

Hyndman does not teach or suggest at least one compartment that is implemented by an operating system, as recited by claim 1. *Hyndman* appears to teach a “building block” (“BB”) that “comprises a database for storing access control data pertinent to said component including all resources accessible to the BB and all users that have the right to use the BB, according to privileges allocated to each user.” Abstract of *Hyndman*. This fails to teach or suggest a compartment implemented by an operating system that defines whether at least one process contained therein is allowed access to particular system resources. Rather, *Hyndman* merely teaches that access rights for a component are stored to a database.

The Final Office Action further asserts on page 14 thereof:

During patent examination the pending claims must be “given their broadest reasonable interpretation consistent with the specification.” *In re Hyatt*, 211 F.3d 1367, 54 USPQ2d 1664, 1667 (Fed. Cir. 2000)... Both a containment tree (data structure) and data building block satisfy the broadest reasonable interpretation of a compartment.

Irrespective of whether a containment tree and data building block satisfy the broadest reasonable interpretation of a “compartment,” claim 1 does not merely recite a “compartment.” Rather, claim 1 recites a compartment that is implemented by an operating system and that defines whether at least one process contained therein is allowed access to particular system resources. As discussed above, *Hyndman*’s building block comprising a database for storing access control data does not provide such a compartment. Moreover, *Hyndman* provides no teaching that its building block is implemented by an operating system.

The Final Office Action further asserts that *Stoecker* teaches a compartment implemented by an operating system, as recited by claim 1. *Stoecker* mentions that TMN standards refer to a containment tree that specifies a relationship between managed objects. However, the containment tree is not taught by *Stoecker* as being a compartment implemented by an operating system.

Claim 1 specifically recites “implementing, by an operating system, at least one compartment for containment of at least one process executable on said processor-based system, wherein said at least one compartment defines whether said at least one process contained therein is allowed access to particular system resources” (emphasis added). Examples of implementing such a compartment for containment by an operating system are described in the specification of the present application at, for instance, page 3, line 6 – page 7, line 28 and page 12, line 16 – page 29, line 16. At best, *Stoecker* teaches an application running on a system (which undoubtedly includes an operating system), where the application implements a containment tree. *Stoecker* makes no mention of its operating system and fails to provide any teaching whatsoever of an operating system that implements a compartment for containment as recited by claim 1. Thus, even if, *arguendo*, the application in *Stoecker* implementing a containment tree is considered as implementing a compartment, *Stoecker* fails to provide any teaching whatsoever of an operating system implementing such compartment (even assuming an operating system is present on *Stoecker*’s system), but instead *Stoecker* expressly teaches an application executing on top of any such operating system (e.g., in application space) implementing such containment tree. Neither of *Stoecker* and *Hyndman* is directed to a trusted operating system, such as the exemplary trusted operating systems described in the specification of the present application.

The Final Office Action asserts on page 14 thereof:

Stoecker teaches implementing at least one compartment for containment (containment tree) at least one process executable on said processor-based system (*col. 5, lines 13-28, etc.*). The computer processor-based system has an operating system. All computer systems have an operating system in order for the processor to perform processing.

However, this merely asserts that *Stoecker* teaches a containment tree and also asserts that all computer systems have an operating system. This fails to even assert that *Stoecker*

teaches that its containment tree is implemented by an operating system. Even assuming that the assertion that all computer systems have an operating system is accurate, no teaching of *Stoecker* is identified that provides that the containment tree is implemented by such an operating system.

Further, neither *Stoecker* nor *Hyndman* teaches or suggests providing at least one operating system command-line utility executable to manipulate the at least one compartment. While these references may provide interfaces to applications, they simply provide no teaching whatsoever of an operating system command-line utility executable to manipulate a compartment. For instance, *Hyndman* teaches “an access control user interface connected to the access control library for viewing and editing the access control data on the GUI” (col. 3, lines 13-15). While *Hyndman* appears to teach such a user interface to a database, *Hyndman* fails to teach an operating system command-line utility executable to manipulate the at least one compartment, as recited by claim 1. Again, *Hyndman* does not teach or suggest a compartment implemented by an operating system, much less one which is manipulatable by a command-line utility. Thus, *Hyndman* does not provide an operating system command-line utility executable to manipulate the at least one compartment.

Similarly, *Stoecker* does not teach or suggest a compartment implemented by an operating system. Thus, *Stoecker* also fails to provide an operating system command-line utility executable to manipulate such a compartment.

Accordingly, the applied combination of *Stoecker* and *Hyndman* fails to teach or suggest all elements of independent claim 1, and thus claim 1 is not obvious under 35 U.S.C. § 103(a) over these references. Therefore, Appellant respectfully requests that the rejection of claim 1 be overturned.

Claims 3, 5, and 7 each depend either directly or indirectly from independent claim 1, and thus inherit all limitations of independent claim 1. It is respectfully submitted that dependent claims 3, 5, and 7 are allowable not only because of their dependency from independent claim 1 for the reasons discussed above, but also in view of their novel claim features (which both narrow the scope of the particular claims and compel a broader interpretation of independent claim 1 from which they depend).

Dependent Claim 8

Dependent claim 8 depends from claim 7, which depends from independent claim 1, and thus claim 8 includes all of the limitations of claims 1 and 7 in addition to its own supplied limitations. It is respectfully submitted that dependent claim 8 is allowable at least because of its dependence from claim 1 for the reasons discussed above.

Claim 7 recites “providing at least one rule that defines containment of said at least one compartment in at least one configuration file.” Claim 8 further recites “providing at least one command-line utility executable to manipulate said at least one rule.” The combination of *Stoecker* and *Hyndman* fails to teach or suggest providing such a command-line utility that is executable to manipulate a rule that defines containment of at least one compartment in a configuration file. Therefore, the rejection of claim 8 should be overturned.

Dependent Claim 9

Dependent claim 9 depends from claim 8, and thus claim 9 includes all of the limitations of claims 1, 7, and 8 in addition to its own supplied limitations. It is respectfully submitted that dependent claim 9 is allowable at least because of its dependence from claims 1 and 8 for the reasons discussed above.

Claim 9 further recites “wherein said at least one command-line utility executable to manipulate said at least one rule comprises at least one command-line utility executable to perform at least one type of rule manipulation selected from the group consisting of: adding a new rule for a particular compartment, removing an existing rule for a particular compartment, and listing all rules for a particular compartment.” The combination of *Stoecker* and *Hyndman* fails to teach or suggest providing such a command-line utility that is executable to perform at least one of the recited types of rule manipulation. Therefore, the rejection of claim 9 should be overturned.

B. Rejection under 35 U.S.C. §103(a) over *Stoecker* in view of *Hyndman* and *Thalhammer-Reyero*

Claim 4 is rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Hyndman* and further in view of U.S. Patent No. 5,930,154 issued to Thalhammer-Reyero (hereinafter “*Thalhammer-Reyero*”). Claim 4 depends from independent claim 1, and thus inherits all limitations of independent claim 1. As discussed above, the combination of *Stoecker* and *Hyndman* fails to teach or suggest all elements of independent claim 1. The Final Office Action does not rely upon *Thalhammer-Reyero* for teaching the above-identified elements of claim 1 that are not taught or suggested by *Stoecker* and *Hyndman*, nor does *Thalhammer-Reyero* provide such teaching. It is therefore respectfully submitted that dependent claim 4 is allowable not only because of its dependency from independent claim 1 for the reasons discussed above, but also in view of its own novel claim features (which both narrows its individual scope and compels a broader interpretation of independent claim 1 from which it depends). Thus, the rejection of claim 4 should be overturned.

C. Rejection under 35 U.S.C. §103(a) over *Stoecker* in view of *Hyndman* and *Tate*

Claim 6 is rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Hyndman* and further in view of U.S. Patent No. 6,493,751 issued to Tate et al. (hereinafter “*Tate*”). Claim 6 depends from claim 5, which depends from independent claim 1, and thus claim 6 inherits all limitations of independent claim 1. As discussed above, the combination of *Stoecker* and *Hyndman* fails to teach or suggest all elements of independent claim 1. The Final Office Action does not rely upon *Tate* for teaching the above-identified elements of claim 1 that are not taught or suggested by *Stoecker* and *Hyndman*, nor does *Tate* provide such teaching. It is therefore respectfully submitted that dependent claim 6 is allowable not only because of its dependency from independent claim 1 for the reasons discussed above, but also in view of its own novel claim features (which both narrows its individual scope and compels a broader interpretation of independent claim 1 from which it depends).

For example, claim 6 recites “wherein said at least one command-line utility is executable to manipulate said at least one compartment without requiring a user to edit said at least one configuration file.” The applied references fail to teach or suggest such a command-line utility that is executable to manipulate a compartment without requiring a user to edit a configuration file. Thus, the rejection of claim 6 should be overturned.

D. Rejections under 35 U.S.C. §103(a) over *Stoecker* in view of *Tate*

Claims 10, 12, 19, and 26-27 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Tate*. Appellant respectfully traverses these rejections below.

Independent Claim 10 and Dependent Claim 12

The combination of *Stoecker* and *Tate* fails to teach or suggest all elements of claim 10. Independent claim 10 recites

A system comprising:

an operating system stored to a computer-readable media, said operating system implementing at least one compartment to which at least one process executable on said system can be associated;

at least one configuration file stored to a computer-readable media, said at least one configuration file defining said at least one compartment; and means for performing management of said at least one compartment without requiring that a user edit said at least one configuration file in which said at least one compartment is defined. (Emphasis added).

Neither *Stoecker* nor *Tate* teach or suggest an operating system implementing a compartment, as recited by claim 10. *Stoecker* mentions that TMN standards refer to a containment tree that specifies a relationship between managed objects. However, the containment tree is not taught as being a compartment implemented by an operating system. *Stoecker* addresses systems and methods for testing of a telecommunications management network (TMN) agent prior to the development, installation and configuration of a TMN manager, *see* col. 1, lines 7-11 of *Stoecker*. While a containment tree may be used in accordance with TMN standards for specifying a relationship between managed objects, *Stoecker* provides no teaching or suggestion of an operating system implementing a compartment, as recited by claim 10. Similarly, *Tate* does not teach or suggest a compartment that is implemented by an operating system.

In response to the above arguments, the Final Office Action merely asserts that:

The computer processor-based system [of *Stoecker*] has an operating system. All computer systems have an operating system in order for the processor perform processing.

Irrespective of the accuracy of the above assertions, they fail to identify any teaching whatsoever in *Stoecker* of an operating system implementing a compartment.

Thus, the combination of *Stoecker* and *Tate* fails to teach or suggest at least this element of claim 10. Accordingly, claim 10 is not obvious under 35 U.S.C. § 103(a) over *Stoecker* in view of *Tate*. Therefore, the rejection of claim 10 should be overturned.

Claim 12 depends from independent claim 10, and thus inherits all limitations of independent claim 10. It is respectfully submitted that dependent claim 12 is allowable at least because of its dependency from independent claim 10 for the reasons discussed above.

Dependent Claim 19

Dependent claim 19 depends from independent claim 10, and thus includes all of the limitations of claim 10 in addition to its own supplied limitations. It is respectfully submitted that dependent claim 19 is allowable at least because of its dependence from claim 10 for the reasons discussed above.

Claim 19 further recites “wherein said means for performing management comprises at least one operating system command-line utility executable to manage said at least one compartment.” The applied combination of references fails to teach or suggest such an operating system command-line utility. Therefore, the rejection of claim 19 should be overturned.

Independent Claim 26 and Dependent Claim 27

The combination of *Stoecker* and *Tate* also fails to teach or suggest all elements of claim 26. Independent claim 26 recites:

A system comprising:
an operating system implementing at least one compartment to which
at least one process executable on said system can be associated;
at least one configuration file defining said at least one compartment;
and
command-line utility executable for performing management of said at least one compartment without requiring that a user edit said at least one

configuration file in which said at least one compartment is defined.
(Emphasis added).

As discussed above with claim 10, the applied combination of *Stoecker* and *Tate* fails to teach or suggest “an operating system implementing at least one compartment to which at least one process executable on said system can be associated”. Additionally, the combination of *Stoecker* and *Tate* also fails to teach or suggest “a command-line utility executable for performing management of said at least one compartment without requiring that a user edit said at least one configuration file in which said at least one compartment is defined”.

Thus, the combination of *Stoecker* and *Tate* fails to teach or suggest at least these elements of claim 26. Accordingly, claim 26 is not obvious under 35 U.S.C. § 103(a) over *Stoecker* in view of *Tate*. Therefore, the rejection of claim 26 should be overturned.

Claim 27 depends from independent claim 26, and thus inherits all limitations of independent claim 26. It is respectfully submitted that dependent claim 27 is allowable at least because of its dependency from independent claim 26 for the reasons discussed above.

E. Rejections under 35 U.S.C. §103(a) over *Stoecker* in view of *Tate* and *Fletcher*

Claims 11, 14, and 29 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Tate* and further in view of U.S. Patent No. 6,009,274 issued to Fletcher et al. (hereinafter “*Fletcher*”). Claims 11, 14, and 29 each depend either directly or indirectly from one of independent claims 10 and 26. As discussed above, the combination of *Stoecker* and *Tate* fails to teach or suggest all elements of independent claims 10 and 26. The Final Office Action does not rely upon *Fletcher* for teaching the above-identified elements of claims 10 and 26 that are not taught or suggested by *Stoecker* and *Tate*, nor does *Fletcher* provide such teaching. It is therefore respectfully submitted that dependent claims 11, 14, and 29 are allowable not only because of their dependency from their respective independent claims for the reasons discussed above, but also in view of their own novel claim features (which both narrow their individual scope and compel a broader interpretation of the respective independent claim from which they depend).

F. Rejections under 35 U.S.C. §103(a) over *Stoecker* in view of *Tate* and *Thalhammer-Reyero*

Claims 13, 15, and 28 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Tate* and further in view of *Thalhammer-Reyero*. Claims 13, 15, and 28 each depend either directly or indirectly from one of independent claims 10 and 26. As discussed above, the combination of *Stoecker* and *Tate* fails to teach or suggest all elements of independent claims 10 and 26. The Final Office Action does not rely upon *Thalhammer-Reyero* for teaching the above-identified elements of claims 10 and 26 that are not taught or suggested by *Stoecker* and *Tate*, nor does *Thalhammer-Reyero* provide such teaching. It is therefore respectfully submitted that dependent claims 13, 15, and 28 are allowable not only because of their dependency from their respective independent claims for the reasons discussed above, but also in view of their own novel claim features (which both narrow their individual scope and compel a broader interpretation of the respective independent claim from which they depend).

G. Rejections under 35 U.S.C. §103(a) over *Stoecker* in view of *Tate* and *Hyndman*

Claims 16-18 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Tate* and further in view of *Hyndman*. Claims 16-18 each depend either directly or indirectly from independent claim 10. As discussed above, the combination of *Stoecker* and *Tate* fails to teach or suggest all elements of independent claim 10. The Final Office Action does not rely upon *Hyndman* for teaching the above-identified elements of claim 10 that are not taught or suggested by *Stoecker* and *Tate*, nor does *Hyndman* provide such teaching. It is therefore respectfully submitted that dependent claims 16-18 are allowable not only because of their dependency from independent claim 10 for the reasons discussed above, but also in view of their own novel claim features (which both narrow their individual scope and compel a broader interpretation of claim 10 from which they depend).

H. Rejections under 35 U.S.C. §103(a) over *Hyndman* in view of *Stoecker*

Claims 20, 22, and 24 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Hyndman* in view of *Stoecker*. Appellant respectfully traverses this rejection below.

Independent Claim 20 and Dependent Claims 22 and 24

The combination of *Hyndman* and *Stoecker* fails to teach or suggest all elements of claim 20. Independent claim 20 recites

A computer-readable medium including instructions executable by a processor, said computer-readable medium comprising:

library of software functions for managing at least one compartment implemented by an operating system, wherein at least one process ~~can be~~ is associated with said at least one compartment and said at least one compartment defines accessibility of resources for said at least one process associated therewith; and

said library of software functions includes at least one command-line utility executable to manipulate said at least one compartment.

Neither *Hyndman* nor *Stoecker* teach or suggest managing at least one compartment implemented by an operating system, as recited by claim 20. As discussed above with claim 10, *Stoecker* mentions that TMN standards refer to a containment tree that specifies a relationship between managed objects. However, the containment tree is not taught as being a compartment implemented by an operating system. *Stoecker* provides no teaching or suggestion of an operating system implementing a compartment, as recited by claim 20.

Similarly, *Hyndman* does not teach or suggest a compartment that is implemented by an operating system. *Hyndman* appears to teach a “building block” that “comprises a database for storing access control data pertinent to said component including all resources accessible to the BB and all users that have the right to use the BB, according to privileges allocated to each user.” Abstract of *Hyndman*. This fails to teach or suggest a compartment implemented by an operating system. Rather, this merely teaches that access rights for a component are stored to a database.

Thus, the combination of *Hyndman* and *Stoecker* fails to teach or suggest at least this element of claim 20. Accordingly, claim 20 is not obvious under 35 U.S.C. § 103(a) over *Hyndman* in view of *Stoecker*. Therefore, the rejection of claim 20 should be overturned.

Claims 22 and 24 each depend either directly or indirectly from independent claim 20, and thus inherit all limitations of independent claim 20. It is respectfully submitted that dependent claims 22 and 24 are allowable not only because of their dependency from

independent claim 20 for the reasons discussed above, but also in view of their novel claim features (which both narrow the scope of the particular claims and compel a broader interpretation of independent claim 20 from which they depend).

I. Rejection under 35 U.S.C. §103(a) over *Hyndman* in view of *Stoecker* and *Thalhammer-Reyero*

Claim 21 is rejected under 35 U.S.C. § 103(a) as being unpatentable over *Hyndman* in view of *Stoecker* and further in view of *Thalhammer-Reyero*. Claim 21 depends from independent claim 20. As discussed above, the combination of *Hyndman* and *Stoecker* fails to teach or suggest all elements of independent claim 20. The Final Office Action does not rely upon *Thalhammer-Reyero* for teaching the above-identified elements of claim 20 that are not taught or suggested by *Hyndman* and *Stoecker*, nor does *Thalhammer-Reyero* provide such teaching. It is therefore respectfully submitted that dependent claim 21 is allowable not only because of its dependency from independent claim 20 for the reasons discussed above, but also in view of its own novel claim features (which both narrows its individual scope and compels a broader interpretation of claim 20 from which it depends).

J. Rejection under 35 U.S.C. §103(a) over *Hyndman* in view of *Stoecker* and *Tate*

Claim 23 is rejected under 35 U.S.C. § 103(a) as being unpatentable over *Hyndman* in view of *Stoecker* and further in view of *Tate*. Claim 23 depends from claim 22 which depends from independent claim 20. As discussed above, the combination of *Hyndman* and *Stoecker* fails to teach or suggest all elements of independent claim 20. The Final Office Action does not rely upon *Tate* for teaching the above-identified elements of claim 20 that are not taught or suggested by *Hyndman* and *Stoecker*, nor does *Tate* provide such teaching. It is therefore respectfully submitted that dependent claim 23 is allowable not only because of its dependency from independent claim 20 for the reasons discussed above, but also in view of its own novel claim features (which both narrows its individual scope and compels a broader interpretation of claim 20 from which it depends).

K. Rejection under 35 U.S.C. §103(a) over *Stoecker* in view of *Hyndman* and *Kuhn*

Claim 25 is rejected under 35 U.S.C. § 103(a) as being unpatentable over *Stoecker* in view of *Hyndman* and further in view of U.S. Patent No. 6,023,765 issued to Kuhn (hereinafter “*Kuhn*”). Claim 25 depends from independent claim 1. As discussed above, the combination of *Stoecker* and *Hyndman* fails to teach or suggest all elements of independent claim 1. The Final Office Action does not rely upon *Kuhn* for teaching the above-identified elements of claim 1 that are not taught or suggested by *Stoecker* and *Hyndman*, nor does *Kuhn* provide such teaching. It is therefore respectfully submitted that dependent claim 25 is allowable not only because of its dependency from independent claim 1 for the reasons discussed above, but also in view of its own novel claim features (which both narrows its individual scope and compels a broader interpretation of claim 1 from which it depends).

For example, claim 25 further recites “wherein said implementing at least one compartment comprises: utilizing a kernel for enforcing said at least one compartment.” As discussed above, the applied references fail to teach or suggest an operating system implementing a compartment. Further, the references fail to teach or suggest utilizing a kernel for enforcing the compartment. Therefore, the rejection of claim 25 should be overturned.

L. Conclusion

In view of the above, Appellant requests that the board overturn the outstanding rejections of claims 1 and 3-29. Attached hereto are a Claims Appendix, Evidence Appendix, and Related Proceedings Appendix. As noted in the attached Evidence Appendix, no evidence pursuant to §§ 1.130, 1.131, or 1.132 or entered by or relied upon by the examiner is being submitted. Also, as noted by the Related Proceedings Appendix, no related proceedings are referenced in II above, and thus no copies of decisions in related proceedings are provided.

I hereby certify that this correspondence is being deposited with the United States Postal Service as Express Mail, Label No. EV 568242729US in an envelope addressed to: M/S Appeal Brief-Patents, Commissioner for Patents, Alexandria, VA 22313.

Date of Deposit: June 28, 2006

Typed Name: Gail L. Miller

Signature: Gail L. Miller

Respectfully submitted,

By: Jody C. Bishop
Jody C. Bishop
Attorney/Agent for Applicant(s)
Reg. No. 44,034
Date: June 28, 2006
Telephone No. (214) 855-8007

VIII. CLAIMS APPENDIX

Claims Involved in the Appeal of Application Serial No. 09/896,385

1. A method of administering a processor-based system, said method comprising: implementing, by an operating system, at least one compartment for containment of at least one process executable on said processor-based system, wherein said at least one compartment defines whether said at least one process contained therein is allowed access to particular system resources; and

providing, by said processor-based system, at least one operating system command-line utility executable to manipulate said at least one compartment.

2. (Canceled)

3. The method of claim 1 wherein said at least one process is labeled to identify the compartment in which it is contained.

4. The method of claim 1 wherein said at least one command-line utility executable to manipulate said at least one compartment comprises at least one command-line utility executable to perform at least one type of compartment manipulation selected from the group consisting of:

adding a new compartment, renaming an existing compartment, removing an existing compartment, resizing an existing compartment, adding a process to a compartment, and removing a process from a compartment.

5. The method of claim 1 wherein said implementing step comprises: defining said at least one compartment in at least one configuration file.

6. The method of claim 5 wherein said at least one command-line utility is executable to manipulate said at least one compartment without requiring a user to edit said at least one configuration file.

7. The method of claim 1 wherein said implementing step comprises: providing at least one rule that defines containment of said at least one compartment in at least one configuration file.

8. The method of claim 7 further comprising the step of:
providing at least one command-line utility executable to manipulate said at least one rule.

9. The method of claim 8 wherein said at least one command-line utility executable to manipulate said at least one rule comprises at least one command-line utility executable to perform at least one type of rule manipulation selected from the group consisting of:

adding a new rule for a particular compartment, removing an existing rule for a particular compartment, and listing all rules for a particular compartment.

10. A system comprising:
an operating system stored to computer-readable medium, said operating system implementing at least one compartment to which at least one process executable on said system can be associated;
at least one configuration file stored to computer-readable medium, said at least one configuration file defining said at least one compartment; and
means for performing management of said at least one compartment without requiring that a user edit said at least one configuration file in which said at least one compartment is defined.

11. The system of claim 10 wherein said means for performing management of said at least one compartment further enables management actions initiated via said means for performing management to be performed dynamically, without requiring that the system be re-booted in order for said management actions to be effective within said system.

12. The system of claim 10 wherein said performing management of said at least one compartment comprises manipulating said at least one compartment.

13. The system of claim 12 wherein said manipulating said at least one compartment includes at least one type of manipulation selected from the group consisting of:
adding a new compartment, renaming an existing compartment, and removing an existing compartment, resizing an existing compartment, adding a process to a compartment, and removing a process from a compartment.

14. The system of claim 12 wherein said means for performing management of said at least one compartment further enables manipulating of said at least one compartment to be performed dynamically, without requiring that the system be re-booted in order for compartment manipulation to be effective within said system.

15. The system of claim 10 wherein said performing management of said at least one compartment comprises switching from a first compartment to a second compartment.

16. The system of claim 10 further comprising:
at least one configuration file including at least one rule defining containment of said at least one compartment.

17. The system of claim 16 wherein said performing management of said at least one compartment comprises manipulating said at least one rule.

18. The system of claim 17 wherein said manipulating said at least one rule comprises at least one type of manipulation selected from the group consisting of:
adding a new rule for a particular compartment, removing an existing rule for a particular compartment, and listing all rules for a particular compartment.

19. The system of claim 10 wherein said means for performing management comprises at least one operating system command-line utility executable to manage said at least one compartment.

20. A computer-readable medium including instructions executable by a processor, said computer-readable medium comprising:

library of software functions for managing at least one compartment implemented by an operating system, wherein at least one process is associated with said at least one compartment and said at least one compartment defines accessibility of resources for said at least one process associated therewith; and

said library of software functions includes at least one command-line utility executable to manipulate said at least one compartment.

21. The computer-readable medium of claim 20 wherein at least one command-line utility executable to manipulate said at least one compartment includes at least one type of manipulation selected from the group consisting of:

adding a new compartment, renaming an existing compartment, and removing an existing compartment, resizing an existing compartment, adding a process to a compartment, and removing a process from a compartment.

22. The computer-readable medium of claim 20 wherein at least one configuration file is implemented on a system to define said at least one compartment.

23. The computer-readable medium of claim 22 wherein said at least one command-line utility is executable to manipulate said at least one compartment without requiring that a user edit said at least one configuration file.

24. The computer-readable medium of claim 20 wherein at least one rule is implemented to define accessibility of resources allowed for said at least one compartment, and wherein said library of software functions further includes at least one command-line utility executable to manipulate said at least one rule.

25. The method of claim 1 wherein said implementing at least one compartment comprises:

utilizing a kernel for enforcing said at least one compartment.

26. A system comprising:

an operating system stored to computer-readable medium, said operating system implementing at least one compartment to which at least one process executable on said system can be associated;

at least one configuration file stored to computer-readable medium, said at least one configuration file defining said at least one compartment; and

command-line utility executable for performing management of said at least one compartment without requiring that a user edit said at least one configuration file in which said at least one compartment is defined.

27. The system of claim 26 wherein said performing management of said at least one compartment comprises manipulating said at least one compartment.

28. The system of claim 27 wherein said manipulating said at least one compartment includes at least one type of manipulation selected from the group consisting of:
adding a new compartment, renaming an existing compartment, and removing an existing compartment, resizing an existing compartment, adding a process to a compartment, and removing a process from a compartment.

29. The system of claim 26 wherein said command-line utility enables manipulating of said at least one compartment to be performed dynamically, without requiring that the system be re-booted in order for compartment manipulation to be effective within said system.

IX. EVIDENCE APPENDIX

No evidence pursuant to §§ 1.130, 1.131, or 1.132 or entered by or relied upon by the examiner is being submitted.

X. RELATED PROCEEDINGS APPENDIX

No related proceedings are referenced in II above, and thus no copies of decisions in related proceedings are provided.

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400



06-29-06

AF ✓
JFW

PATENT APPLICATION

ATTORNEY DOCKET NO. 10013499-1

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Joubert Berger et al.

Confirmation No.: 9535

Application No.: 09/896,385

Examiner: K. Tang

Filing Date: June 29, 2001

Group Art Unit: 2195

Title: SYSTEM AND METHOD FOR MANAGEMENT OF COMPARTMENTS IN A TRUSTED OPERATING SYSTEM

Mail Stop Appeal Brief-Patents
Commissioner For Patents
PO Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF

Transmitted herewith is the Appeal Brief in this application with respect to the Notice of Appeal filed on April 28, 2006.

The fee for filing this Appeal Brief is (37 CFR 1.17(c)) \$500.00.

(complete (a) or (b) as applicable)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136(a) apply.

☐ (a) Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d)) for the total number of months checked below:

☐ 1st Month
\$120

☐ 2nd Month
\$450

☐ 3rd Month
\$1020

☐ 4th Month
\$1590

☐ The extension fee has already been filed in this application.

☒ (b) Applicant believes that no extension of time is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Please charge to Deposit Account 08-2025 the sum of \$ 500. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16 through 1.21 inclusive, and any other sections in Title 37 of the Code of Federal Regulations that may regulate fees. A duplicate copy of this sheet is enclosed.

☒ I hereby certify that this correspondence is being deposited with the United States Postal Service as Express Mail, Label No. EV 568242729US addressed to: Commissioner for Patents, Alexandria, VA 22313-1450
Date of Deposit: June 28, 2006

OR

☐ I hereby certify that this paper is being transmitted to the Patent and Trademark Office facsimile number (571)273-8300.

Date of facsimile:

Typed Name: Gail L. Miller

Signature: Gail L. Miller

Respectfully submitted,

Joubert Berger et al.

By Jody C. Bishop

Jody C. Bishop

Attorney/Agent for Applicant(s)

Reg No.: 44,034

Date: June 28, 2006

Telephone: (214) 855-8007